AMENDMENTS TO THE SPECIFICATION

Applicant respectfully requests the following changes be made to the specification. The following amendments correct typographical errors and render the application's internal cross references consistent including references to Figure 4 as amended (see below).

Please replace the paragraph beginning at Page 10, Line 12 with the following paragraph:

FIGURE 2B is a block diagram showing additional details of softswitch processes and storage 56. Softswitch processes and storage 56 includes a plurality of processes 64. Each process may be associated with a plurality of calls 68. Softswitch processes and storage 56 also includes a plurality of buffers 66. Each buffer 66 may have a plurality of associated memory locations 72. In one implementation, one call 68 may consume two memory locations 72; however, other implementations may require more or less memory locations associated with each call 68. Because, in some embodiments, the amount of memory usage is related to the number of calls being processed by softswitch 12, as shown in ~~FIGURE 7B~~FIGURE 2B, the number of calls being processed by the softswitch may be indicative of a potential failure of computer 11. Thus, the number of calls being processed by the softswitch may also be monitored to prevent overload.

Please replace the paragraph beginning at Page 12, Line 9 with the following paragraph:

In addition to monitoring memory usage, according to one embodiment the processor usage is also monitored. At a step 108 the usage of processor system 46 is checked. This may occur, in one example, by separate checks of the processor usage of processor 52 and processor 54. One method for implementing such a check of processor usage is the UNIX shell command "VMSTAT", offered by Sun Microsystems. This UNIX shell command determines and provides the processor usage periodically. According to one embodiment of the invention, it has been determined that checking approximately every ten seconds is desirable; however, other intervals may be utilitized. Thus the amount of processor usage may be implemented by using a time interval option of the shell command VMSTAT. At step 110 a determination is made of whether the processor usage exceeds acceptable limits. This determination may be made for either processor 52 or 54, or both. Any desired limit may be used; however, according to one embodiment of the invention it has been determined that when the processor usage exceeds 90-95% of capacity that remedial action should be taken. Thus, at step 110 if the usage limit is exceeded, flow control module 76 is executed at ~~step 106~~step 116. If the usage limit is not exceeded, CPU usage is checked periodically at step 108.

Please replace the paragraph beginning at Page 13, Line 16 with the following paragraph:

In addition to checking the memory usage at step 102 and the processor usage at step 108, the number of outstanding calls for each signaling subsystem may be monitored as an indication of potential softswitch overload. At step 112, the number of outstanding calls for each signaling subsystem is determined. This step may be performed in a variety of ways; however, according to one embodiment, a counter is maintained for each signaling subsystem. Whenever a call is placed a counter is incremented and is not decremented until the call is terminated. At a step 114, a comparison is made, for each signaling subsystem, between the number of calls outstanding and the number of calls for each subsystem that may take place. As illustrated in FIGURE 2B, each call is associated with a particular number of buffers, which results in the limitation on the number of calls when the available buffers for each signaling subsystem are used. Typically, an established call 68 uses two of the buffers 72 in softswitch storage 66. One buffer is used for the originating leg and one for the terminating leg. According to one embodiment, when the number of calls, whether in a connecting, established or disconnecting stage, exceeds 50% of the maximum capacity of buffers, this is considered excessive and flow control ~~module 106~~module 116 is executed. If the number of calls at step 114 is not exceeded, then step 112 is periodically executed.